Applied Machine Learning with Big Data
"EE 6973"

Paul Rad, Ph.D.
Chief Research Officer
UTSA Open Cloud Institute(OCI)
University of Texas at San Antonio

Deep Learning Best Practices

Bias and Variance

Training Data Set **Test Data Set**

Human Level Error = 5%
Training Error = 30%
Test Error = 33%

What to do next ????

Bigger Model
Training Longer or Use GPU (Faster Systems)
New Model architecture

Human Level Error = 5%
Training Error = 7%
Test Error = 30%

What to do next ????

"Over fitting problem"

More Data
Regularization
New Model architecture

End-to-End Deep Learning

Simple (Classification or Regression)

- Movie Review → Sentiment
- Image → Object Recognition

"DL, CNN, RNN" models

2nd Major Trends

- Image → Caption
- Audio → Transcript
- English → French
- Parameter → image

K-Means Clustering

3 Types of Learning

Supervised

- Learning from labeled data
- E.g., Spam classification

- Classification
- Regression
- Ranking

Unsupervised

- Discover structure in unlabeled data
- E.g., Document clustering

- Clustering
- Hidden Markov Models

Reinforcement

- Learning by “doing” with delayed reward
- E.g., Chess computer

What is Clustering?

Attach label to each observation or data points in a set

You can say this "unsupervised classification"

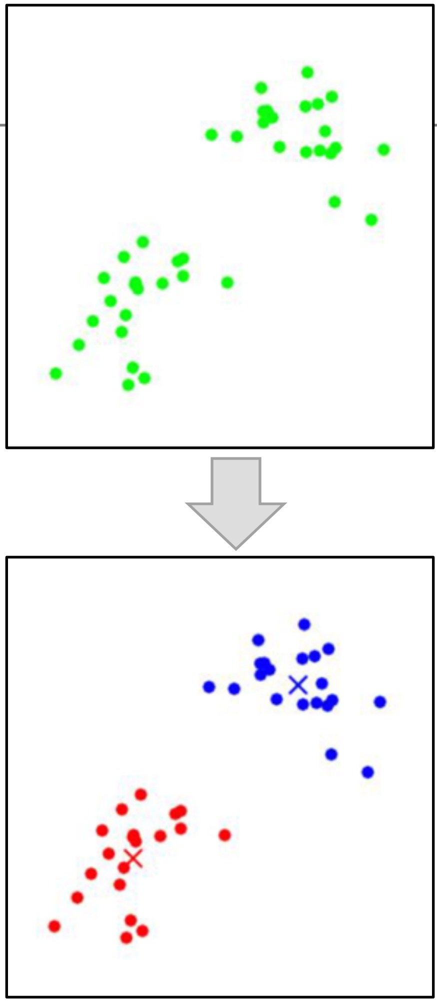
Clustering is alternatively called as "grouping"

Intuitively, if you would want to assign same label to a data points that are "close" or "similar" to each other. Thus, clustering algorithms rely on a distance metric between data points

For example: Euclidean distance

$$E = \Sigma_{k=1}^{K} \Sigma_{\mathbf{x} \in C_k} d^2(\mathbf{x}, \mathbf{m}_k)$$

$$d^2(\mathbf{x}, \mathbf{m}_k) = \sum_{n=1}^{N} (x_n - m_{kn})^2$$



NP-hard combinatorial optimization problem

In how many ways can we assign *K* labels to *N* observations?

For each such possibility, we can compute a cost. Pick up the assignment with best cost.

Number of possible classes:

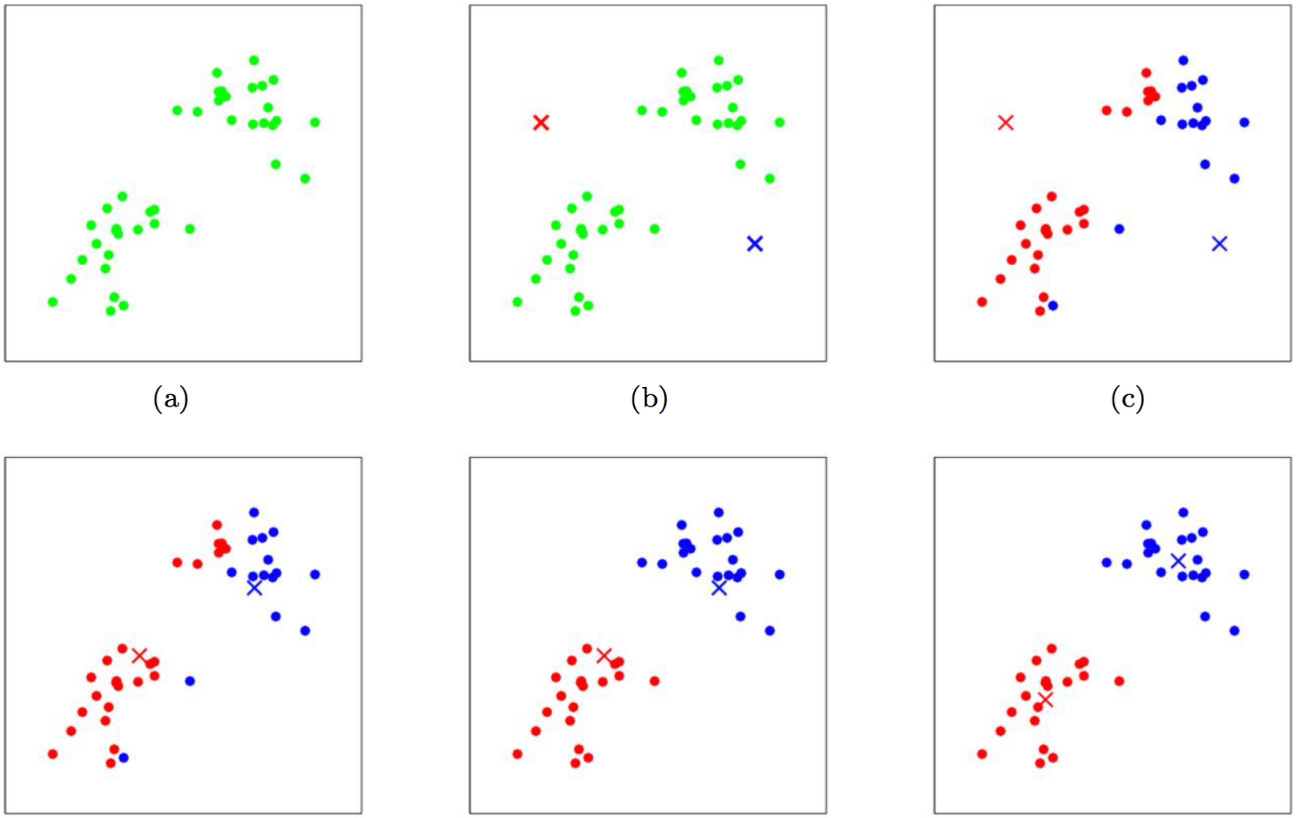
$$S(N,K) = \frac{1}{K!}\sum_{k=1}^{K}(-1)^{K-k}\binom{K}{k}k^N$$

What is K-means Clustering?

k-means clustering aims to partition n observations into **k clusters** in which each observation belongs to the **cluster** with the nearest **mean**, serving as a prototype of the **cluster**.

- An unsupervised clustering algorithm
- "*K*" stands for number of clusters, it is typically a user input to the algorithm; some criteria can be used to automatically estimate K. It is an approximation to an NP-hard combinatorial optimization problem but Easy to implement.
- *K*-means algorithm is iterative in nature and works only for numerical data

Example



(a) (b) (c) (d) (e) (f)

K-Means

The *K-means* algorithm: a heuristic method

- K-means algorithm: each cluster is represented by the center of the cluster and the algorithm converges to stable centroids of clusters.

- K-means algorithm is the simplest but computationally expensive partitioning method for clustering analysis

K-Means

Given the cluster number K, the K-means algorithm is carried out in three steps after initialization:

Initialization: randomly set seed points

1) Assign each object to the cluster of the nearest seed point measured with a specific distance metric

2) Compute new seed points as the centroids of the clusters of the current partition (the centroid is the canter, i.e., *mean point*, of the cluster)

3) Go back to Step 1), stop when no more new assignment (i.e., membership in each cluster no longer changes)

K-Means Algorithm

Randomly initialize K cluster centroids C1, C2, …, Ck

Repeat {

for i= 1 to N

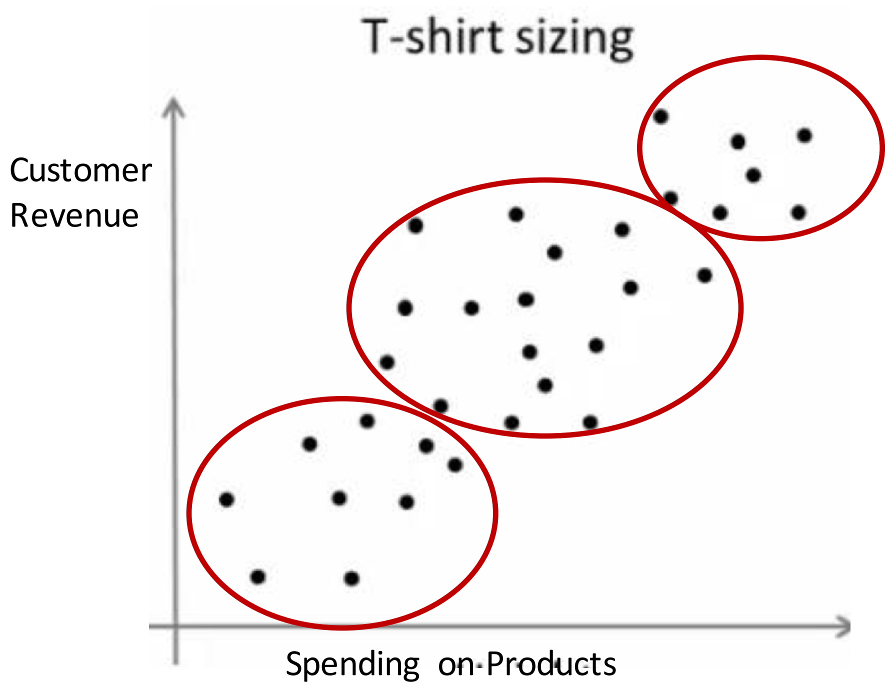
xi := Label as the closest cluster centroid

for k= 1 to k

Ck := mean of points assigned to cluster k

}

Example: Customer Segmentation based on spending and revenue "T-shirt sizing"



T-shirt size problem